Sample commands for ChIP-seq & nucleosome calling pipeline:

Commands

Bowtie

```
bowtie2 -x $GENOME -1 $SAMPLE1 -2 $SAMPLE2 -S $GENOME_OUTPUT/$OUTPUT_NAME
```

BWA

```
bwa mem -t 4 -R '@RG\tID:M1_1_L3\tSM:1_L3\tLB:L3' $GENOME $SAMPLE1 $SAMPLE2 | \ samtools
view -Sb - > $OUTPUT.bam
```

Post-Processing (samtools)

```
# Sam to Bam
samtools view -Sb $SAMPLE > ${S%.*}.bam

## Sorting
samtools sort $SAMPLE ${S%.*}.sorted

## Remove Dups
samtools rmdup $SAMPLE ${S%.*}.rmdup.bam

## Create BAI index
samtools index $SAMPLE

## Mapping Stats
samtools flagstat $SAMPLE > ${S%%.*}_mappings.txt

## Merge Samples
samtools merge $OUTPUT $SAMPLE1 $SAMPLE2
```

Peak Calling (MACS)

```
macs2 callpeak -t $BASEDIR/$SAMPLE -c $BASEDIR/$INPUT_1 $BASEDIR/$INPUT_2 -f BAMPE -g
2.7e+9 -n $BASEDIR/$OUTPUT_NAME --tempdir $TEMPDIR --call-summits -B
```

NucleR

```
library(nucleR)
# Read in BAM
sample <- readBamGappedAlignmentPairs(sample.bam)
input <- readBamGappedAlignmentPairs(input.bam)

# Calculate the coverage, directly in reads per million (r.p.m)
cover_sample = coverage(sample)
cover_input = coverage(input)
```

```
# Control correction (ChIP:Input)
corrected = controlCorrection(cover_sample, cover_input, mc.cores=1)

# Call nucleosomes (MNase bias cleaning)
# Use peaks from FASTA files
clean_corrected =filterFFT(corrected[[“chrN”]][start:end], pcKeepComp=0.02)

# Call peaks
peaks_corrected = peakDetection(clean_corrected, width =135,threshold="95%", score= TRUE)

# Plot Peaks
plotPeaks(peaks_corrected, clean_corrected, threshold="95%", ylab="coverage",
indiv.scores=FALSE)

# Identify "fuzzy" Nucleosomes and extract locations
# pos = 'start'position when calling 'clean_corrected' as initial position is represented
as 0
NucPeaks <- function(peaks_corrected, pos, chrom, name){
    nuc_calls <- ranges(peaks_corrected)
    red_calls <- reduce(nuc_calls)
    red_class <- RangedData(red_calls)
    data <- as.data.frame(red_class)
    data <- data[data$width <= 130,]
    data[,2] <- data[,2] + pos
    data[,3] <- data[,3] + pos
    data[,1] <- chrom
    data <- data[,1:3]
    write.table(data,name,sep='\t')
}
```

- `coverage`: counts the number of times a position is represented in a set of ranges
- `controlCorrection`: Allows for correction of experimental coverage profiles of MNase digested nucleosomal DNAs with control samples (ChIP input)
- `filterFFT`: Literature usually smoothes signal using a sliding window average and then uses a HMM to determine the probabilities of having one or another state. However, short windows allow for too much noise while larger windows change the position and shape of the peaks. Therefore, a Fourier Analysis is used for filtering of the signal and selection of its principal components. The coverage values are converted to the Fourier Space and components are knocked out which are greater than the given percentile in order to remove noise from the profile
- `peakDetection`: Since FFT is applied, this function calls peaks by viewing changes in the trends of the profile.
- Peak scores are relative to the height (direct measure of coverage) and sharpness (how fuzzy a nucleosome is). If a peak is very narrow and the surrounding regions are depleted, this is an indicator of a good positioned nucleosome
- `width`: If a positive integer >1 is given, the peaks are returned as a range of the given width centered in the local maximum

File clean up to make BED file

```
For i in *.txt
do
sed -i '1d' $i
done

cat *.txt > all_chr_peaks.txt

awk '{print $2, $3, $4}' all_chr_peaks.txt > all_chr_peaks.bed

sed -i 's/"//g' all_chr_peaks.bed
```

Bedtools

```
# Get FASTA
bedtools getfasta -fi $GENOME -bed all_chr_peaks.bed -fo all_chr_peaks.fasta
```